AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Previously Presented) A method executed in a data processing system for providing communication access between a first process associated with a first node and a second process associated with a second node, the method comprising:

sending a request from the first node to an administrative machine to verify a first node identification associated with the first process;

in response to the request, receiving security context information at the first node from the administrative machine, the security context information comprising a virtual address for the first node;

appending the security context information for the first process in a process table, the process table listing a first process identifier associated with the first process executing in memory;

opening a socket between the first process and the second process;

transmitting a packet from the first process to the second process through the open socket without passing through the administrative machine, only after determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, the packet comprising the security context information for the first process in the process table, each said channel comprising a collection of virtual links through a public network infrastructure; and

receiving the transmitted packet.

2. (Original) The method of claim 1, further comprising modifying a socket structure so as to accept the security context information.

3. (Original) The method of claim 1, further comprising:

receiving the packet at the second process through the socket;

verifying the security context information received in the packet; and

permitting use of the packet if the security context information is verified.

4. (Canceled).

5. (Previously Presented) The method of claim 1, wherein determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway comprises:

comparing the security context information in the packet and security context information in another process table.

6. (Canceled).

7. (Previously Presented) The method of claim 3, wherein verifying the security context information comprises:

determining whether the first and second process belong to two different linked channels; and

permitting use of the packet when the different channels are linked.

8. (Previously Presented) The method of claim 7, wherein determining whether the first and second process belong to two different linked channels comprises:

initiating a process that spawns two child processes that are connected by a shared-memory region in a memory.

9. (Previously Presented) The method of claim 7, wherein permitting use of the packet comprises:

decrypting the packet; and

authenticating a sender associated with the first process.

10. (Previously Presented) The method of claim 1, wherein appending security context information comprises:

obtaining the security context information from a third process, the security context information comprising a virtual address and a node identification.

11. (Original) The method of claim 1, further comprising:

modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.

12. (Canceled).

13. (Previously Presented) The method of claim 1, wherein receiving security context information further comprises:

receiving a key that corresponds to the first node identification from the server.

14. (Previously Presented) The method of claim 13, further comprising:

encrypting a packet transmitted by the first process using the key;

encapsulating the encrypted packet with a header that comprises the first node identification.

15. (Previously Presented) The method of claim 1, further comprising:

sending a second request from the second node to the server to verify a second node identification;

receiving additional security context information from the server, wherein the additional security context information comprises a second virtual address for the second node;

creating the second process; and

appending the security context information for the second process in the process table associated with the second process.

16. (Previously Presented) A method executed in a data processing system for providing secure communications between a first process associated with a first node and a second process associated with a second node, the method comprising:

obtaining a node identification comprising a virtual address from an administrative machine;

including the node identification in a field corresponding to the first process in a process table, the process table listing a first process identifier associated with the first process executing in memory;

transmitting, only after determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, a datagram that contains the node identification from the first process to a socket, each said channel comprising a collection of virtual links through a public network infrastructure;

receiving the datagram at the second process that contains the node identification and a second virtual address, without the datagram passing through the administrative machine.

17. (Previously Presented) The method of claim 16, wherein obtaining a node identification further comprises:

modifying a socket structure in the socket so that the socket structure accepts the node identification; and

modifying a process table so that the table comprises a node identification field.

18. (Currently Amended) A system for providing communication access between a first process associated with a first node corresponding to a first computer system and a second process associated with a second node corresponding to a second computer system networked via a public network infrastructure to the first computer system, the system including the first computer system and the second computer system, the first computer system comprising:

means for sending, across the public network, a request from the first node to an administrative machine ~~a server~~ associated with a private network to verify a first node identification associated with the first process;

means for receiving security context information, in response to the request, at the first node from the administrative machine, the security context information comprising a virtual address for the first node;

means for appending the security context information for the first process in a process table, the process table listing a first identifier associated with the first process executing in memory;

means for opening a socket between the first process and the second process;

means for determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, each said channel comprising a collection of virtual links through a public network infrastructure;

means for transmitting a packet from the first process to the second process through the open socket without passing through the administrative machine, the packet comprising the security context information for the first process in the process table; and

means for receiving the transmitted packet.

19. (Original) The system of claim 18, further comprising means for modifying a socket structure so as to accept the security context information.

20. (Previously Presented) The system of claim 18, further comprising:

means for verifying the security context information received in the packet; and

means for permitting use of the packet if the security context information is verified.

21. (Canceled).

22. (Previously Presented) The system of claim 18, wherein means for determining if the first and second process belong to a channel comprises:

means for comparing the security context information in the received packet and security context information in another process table.

23. (Canceled).

24. (Previously Presented) The system of claim 20, wherein means for verifying the security context information comprises:

means for determining whether the first and second process belong to two different linked channels; and

means for permitting use of the packet when the different channels are linked.

25. (Previously Presented) The system of claim 24, wherein means for determining whether the first and second process belong to two different linked channels comprises:

means for initiating a process that spawns two child processes that are connected by a shared-memory region in a memory.

26. (Previously Presented) The system of claim 24, wherein means for permitting use of the packet comprises:

means for decrypting the packet; and

means for authenticating a sender associated with the first process.

27. (Previously Presented) The system of claim 18, wherein means for appending security context information comprises:

means for obtaining the security context information from a third process, the security context information comprising a virtual address and a node identification.

28. (Original) The system of claim 18, further comprising:

means for modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.

29. (Previously Presented) A system for placing a process executed in a node in a security context, comprising:

an administrative machine; and

a sending node comprising:

a transmission module that transmits a request to the administrative machine to verify a sending node identification, and receives security context information from the administrative machine in response to the request, wherein the security context information comprises a virtual address for the sending node;

memory containing a process and an associated process table, the process table listing a first process identifier associated with the first process executing in memory;

an appending module that appends the received security context information and the sending node identification for the process in the process table, wherein the transmission module transmits a packet from the process to a receiving node without passing through the administrative machine, only after determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, the packet comprising the security context information for the first process in the process table, each said channel comprising a collection of virtual links through a public network infrastructure; and

means for receiving the transmitted packet.

30. (Previously Presented) The system of claim 29, wherein the transmission module further receives a key that corresponds to the sending node identification from the administrative machine.

31. (Previously Presented) The system of claim 30, further comprising: an encryption module that encrypts the packet transmitted by the process using the key; and

an encapsulating module that encapsulates the encrypted packet with a header that comprises the sending node identification.

32. (Canceled).

33. (Currently Amended) A system for providing secure communications between a first process associated with a first node corresponding to a first computer system and a second process associated with a second node corresponding to a second computer system networked via a public network infrastructure to the first computer system, the system including the first computer system and the second computer system, the first computer system comprising:

means for obtaining a node identification comprising a virtual address from an administrative machine connected to the first computer system via the public network infrastructure;

means for including the node identification in a field corresponding to the first process in a process table, the process table listing a first process identifier associated with the first process executing in memory;

means for transmitting a datagram that contains the node identification from the first process to a socket;

means for determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, each said channel comprising a collection of virtual links through a public network infrastructure;

means for receiving the datagram at the second process that contains the node identification and a second virtual address, without the datagram passing through the administrative machine; and

means for accepting the transmitted packet.

34. (Previously Presented) The system of claim 33, wherein means for obtaining a node identification further comprises:

means for modifying a socket structure in the socket so that the socket structure accepts the node identification; and

means for modifying a process table so that the table comprises a node identification field.

35. (Currently Amended) A computer readable storage medium installable on a networked computer system in a data processing system, wherein the computer-readable storage medium includes a plurality of modules having a set of instructions which when executed by a processor of the computer system operate to control the ~~for controlling a~~ data processing system to perform a method for providing communication access between a first process associated with a first node disposed on the computer system and a second process associated with a second node, the modules comprising:

a sending module for sending a request from the first node to an administrative machine to verify a first node identification associated with the first process;

a receiving module for receiving security context information, in response to the request, at the first node from the administrative machine, the security context information comprising a virtual address for the first node;

an appending module for appending security context information for the first process in a process table, the process table listing a first process identifier associated with the first process executing in memory;

an opening module for opening a socket between the first process and the second process;

a transmitting module for transmitting a packet from the first process to the second process through the open socket without passing through the administrative machine, only after determining that the first process and the second process are connected by at least one of (i) a channel and (ii) a plurality of channels linked by a gateway, the packet comprising the

security context information for the first process in the process table, each said channel comprising a collection of virtual links through a public network infrastructure; and

a receiving module for receiving the transmitted packet.

36. (Original) The computer readable medium of claim 35, further comprising a modifying module for modifying a socket structure so as to accept the security context information.

37. (Original) The computer readable medium of claim 35, further comprising:

a receiving module for receiving the packet at the second process through the socket;

a verifying module for verifying the security context information received in the packet; and

a permitting module for permitting use of the packet if the security context information is verified.

38. (Canceled).

39. (Currently Amended) The computer readable medium of claim 41 ~~48~~, wherein the determining module comprises:

a comparing module that compares the security context information in the received packet and security context information in another process table.

40. (Canceled).

41. (Previously Presented) The computer readable medium of claim 37, wherein the verifying module comprises:

a determining module for determining whether the first and second process belong to two different linked channels; and

a permitting module for permitting use of the packet when the different channels are linked.

42. (Previously Presented) The computer readable medium of claim 41, wherein the determining module comprises a initiating module that initiates a process that spawns two child processes that are connected by a shared-memory region in a memory.

43. (Previously Presented) The computer readable medium of claim 41, wherein the permitting module comprises:

a decrypting module for decrypting the packet; and

an authenticating module for authenticating a sender associated with the first process.

44. (Previously Presented) The computer readable medium of claim 35, wherein the appending module comprises:

an obtaining module for obtaining the security context information from a third process, the security context information comprising a virtual address and a node identification; and

a limiting module for limiting each of the first, second and third processes to communicate with another process provided that the communicating processes share the same node identification.

45. (Original) The computer readable medium of claim 35, further comprising:

a modifying module for modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.

46-48. (Cancelled).